Introducing QoS support in Bluetooth Piconetwith a Class-Based EDF Scheduling

Antoine Mercier, Pascale Minet, Laurent George

To cite this version:

Antoine Mercier, Pascale Minet, Laurent George. Introducing QoS support in Bluetooth Piconetwith a Class-Based EDF Scheduling. [Research Report] RR-5054, INRIA. 2003. inria-00071529

HAL Id: inria-00071529
https://hal.inria.fr/inria-00071529

Submitted on 23 May 2006

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Introducing QoS support in Bluetooth Piconet with a Class-Based EDF Scheduling

Antoine MERCIER — Pascale MINET — Laurent GEORGE

N° 5054

December 2003

THÈME 1

Rapport de recherche

ISSN 0249-6399 ISRN INRIA/RR--5054--FR+ENG

Introducing QoS support in Bluetooth Piconet with a Class-Based EDF Scheduling

Antoine MERCIER[*] , Pascale MINET[†] , Laurent GEORGE[‡]

Thème 1 — Réseaux et systèmes
Projet HIPERCOM

Rapport de recherche n° 5054 — December 2003 — 25 pages

Abstract: In this paper, we focus on the Bluetooth wireless network, analyzing its ability to support Quality of Service (QoS) requirements defined by the application. In particular, we are interested in two QoS parameters: (i) an application constraint denoting the importance degree of a message, and (ii) its delivery deadline. The QoS perceived by the application depends on the efficiency of the scheduling schemes chosen at the medium access layer. We define the minimal knowledge level required by a scheduling scheme to support these QoS constraints. As an example of classical scheduling schemes, we analyze performances of One-Round Robin (1-RR) and show that it does not provide a sufficient service differentiation. To achieve better service differentiation, we first present enhancements accounting locally for the two QoS parameters. These enhancements are applied to 1-RR scheduling scheme and we then give a comparison between the two versions. These comparisons are done by evaluating in each class, the average message response time and the percentage of messages missing their deadline. We then introduce enhancements in the intra-piconet scheduling. So, we define a new Bluetooth global scheduling, called Class-Based Earliest Deadline First (CB-EDF) that takes into account both locally and globally these two QoS parameters. Simulation results show that CB-EDF achieves a good service differentiation and allows the coexistence of messages with different application constraints on the same ACL link. Moreover, CB-EDF is a flexible solution that adapts itself to the provided knowledge level.

Key-words: Bluetooth, QoS, piconet scheduling, service differentiation, class, deadline, knowledge level.

[*] Ecole Centrale d'Electronique, LACCSC, 53 rue de Grenelle, 75007 Paris, *mercier@ece.fr*
[†] INRIA Rocquencourt, 78153 Le Chesnay Cedex, *pascale.minet@inria.fr*
[‡] Université Paris 12, LIIA, 120 rue Paul Armangot, 94400 Vitry, *george@univ-paris12.fr*

Unité de recherche INRIA Rocquencourt
Domaine de Voluceau, Rocquencourt, BP 105, 78153 Le Chesnay Cedex (France)
Téléphone : +33 1 39 63 55 11 —Télécopie : +33 1 39 63 53 30

Support de Qualité de Service pour Bluetooth : ordonnancement à base de classes et d'échéances

Résumé : Nous nous intéressons, dans ce papier, à l'aptitude du réseau sans-fil Bluetooth à supporter les exigences de Qualité de Service exprimées par les applications. Nous nous concentrons en particulier sur deux paramètres de Qualité de Service (QoS): (i) une contrainte indiquant le degré d'importance d'un message, et (ii) une échéance de remise du message au destinataire. La Qualité de Service perçue par l'application dépend de l'efficacité des mécanismes d'ordonnancement choisis au niveau de la couche d'accès au médium. Nous définissons la notion de niveau de connaissance minimal nécessaire à un mécanisme d'ordonnancement pour tenir compte de telles contraintes. A titre d'exemple, nous analysons les performances de 1-Round-Robin (1-RR) et montrons que cet algorithme n'apporte pas de différenciation de service appréciable. Dans le but d'offrir une meilleure différenciation de service, nous présentons, dans un premier temps, une amélioration visant à prendre en compte localement les deux contraintes de QoS précédentes. Cette amélioration est appliquée à l'algorithme 1-RR. Nous donnons des résultats de simulation permettant la comparaison entre la version originale et la version améliorée de 1-RR. Ces simulations sont réalisées en évaluant dans chaque classe, le temps de réponse moyen des messages ainsi que le pourcentage de messages ne respectant pas leur échéance. Nous introduisons ensuite des améliorations dans l'ordonnancement intra-piconet. L'ensemble de ces améliorations constitue un nouvel algorithme d'ordonnancement pour Bluetooth, appelé Class-Based Earliest Deadline First (CB-EDF), qui tient compte localement et globalement des deux contraintes de QoS. Les résultats de simulation montrent que CB-EDF offre une bonne différenciation de service et permet la coexistence de messages avec des contraintes différentes sur un même lien ACL. De plus, CB-EDF est une solution modulaire qui s'adapte au niveau de connaissance fourni.

Mots-clés : Bluetooth, QoS, ordonnancement sur un piconet, différenciation de service, classe, échéance, niveau de connaissance.

Contents

1 Introduction

Originally designed as a simple wireless solution to connect electronic devices, Bluetooth has gained a lot of consideration and attention by the scientific community. Bluetooth [1] is defined as a solution for ad-hoc networks using polling scheme as medium access control. Bluetooth devices are organized into *piconets* where a central device acts as the master, which can manage up to seven active slaves. A Bluetooth device can participate in more than one piconet but can be the master of only one piconet. A set of piconets interconnected by devices that participate in several piconets is referred to as a *scatternet*.

In this paper, we study how Bluetooth devices account for *Quality of Service* (QoS) requirements, expressed by QoS sensitive applications, such as multimedia or sensor/actuator applications [2]. More precisely, we focus on the scheduling mechanisms used to support these QoS requirements. We distinguish three message scheduling mechanisms:

- *Local scheduling* refers to the mechanism selecting one message in the local waiting queues of the Bluetooth device considered. On a slave, the local waiting queue is denoted $S \rightarrow M$. On the master, the local waiting queues are denoted $M \rightarrow S$, one per slave.
- *Intra-piconet scheduling* refers to the polling scheme within a piconet. It is defined in [3] as the set of rules that determines when the piconet master switches from one slave to another. The master solicits packet transmissions from attached slaves. The combination of local and intra-piconet schedulings is referred to as *global scheduling*.
- *Inter-piconet scheduling* is the mechanism that determines the piconet where a bridging device has to be present at a given time in a scatternet configuration. As we focus on a single piconet, inter-piconet scheduling is not considered in this paper.

In order to support QoS applications, real-time constraints must be accounted for by both local and intra piconet schedulings. Both schedulings have a major importance in the performances offered to traffics within a piconet. Precisely, we focus on the support of two QoS application constraints:

- the first one denoting the importance degree of a message, and
- the second referring to the message delivery deadline.

We introduce in this paper the notion of *knowledge level* that expresses the minimal knowledge used by the system to schedule messages. The knowledge levels can range from a null knowledge to the knowledge of an omniscient observer. In section 2, we propose a classification based on the knowledge level required by the global piconet scheduling schemes. In section 3, we show that native One-Round Robin (1-RR) is unable to provide service differentiation. Indeed:

- High importance messages have to wait the end of transmission of low importance messages,
- Messages of the same importance with close deadlines have to wait the end of transmission of messages with far deadlines.

We then propose to introduce local QoS management in 1-RR. The QoS parameters are accounted for by the local scheduling with a combination of class based *Priority Queuing* (PQ) - a class contains all messages of the same importance degree - and *Earliest Deadline First* (EDF). PQ is used between classes and EDF is used within a class. This local improvement allows a better service differentiation. However, it does not guarantee that messages belonging to the highest priority class are transmitted first.

To achieve that, classes and deadlines have to be globally accounted for by the medium access protocol: this function is performed in the intra-piconet scheduling. We then design a global scheduling mechanism that supports, at the global level, the two previous application constraints. This new mechanism is called *Class-Based Earliest Deadline First* (CB-EDF). We propose a modular solution that can be applied with different knowledge levels. Indeed, the algorithm used by a module to perform a given functionality is refined according to the available knowledge. Assuming the highest knowledge level (i.e. omniscient observer), performances obtained by this CB-EDF are given to validate our QoS management principles. In section 4, we show how CB-EDF can cope with realistic knowledge levels by means of a combination of signalling and prediction schemes. In section 5, we study the impact of the prediction scheme and the knowledge level on CB-EDF performances. All performances are evaluated by means of simulations for various scenarios. We compute the average response time of any flow, the percentage of messages missing their deadline and the efficiency of the polling scheme. Finally, we conclude the paper.

2 State of the art

In this section, we present a brief state of the art of local and intra-piconet schedulings. For intra-piconet scheduling, we give a classification of the major polling schemes accounting for the knowledge level they require.

2.1 Preliminaries

Two types of links are defined in a Bluetooth piconet [1]: *Synchronous Connection Oriented* (SCO) links for synchronous traffic (e.g. voice) and *Asynchronous ConnectionLess* (ACL) links. The number of SCO links is limited to three per piconet. Moreover, a SCO link has no flexibility. Hence, we consider only ACL links. ACL links use the available bandwidth left by SCO links. We recall that the master maintains at least one queue per slave, denoted $M \rightarrow S$ and each slave maintains at least a slave-to-master queue, denoted $S \rightarrow M$. The

Bluetooth communication scheme for ACL packets differs from classical schemes mainly for two reasons [3]:

- A $M \rightarrow S$ transmission is immediately followed by a $S \rightarrow M$ transmission. Therefore, the polled slave can always communicate after a master transmission if desired. So, the states of $M \rightarrow S$ and $S \rightarrow M$ queues are strictly related.
- As the polling scheduling is entirely handled by the master, the knowledge of the queues status is only partial. The master only knows the status of all $M \rightarrow S$.

2.2 State of the art of local scheduling

The Bluetooth specifications [1] recommend the use of FIFO queues for the message transmission. The messages are then transmitted by increasing order of their arrival time. A FIFO scheduling does not take into account deadline constraints. No improvement on local queue management is suggested.

We preconize here the use of classes and deadlines to account for the message importance and deadline. The local scheduling is then class based. Moreover, we recommend the use of EDF algorithm within each class, as it has been proved optimal in [4] for a uniprocessor scheduling in both preemptive and non-preemptive context when the message arrival times are not known a priori. EDF optimality means that if EDF does not give a feasible scheduling, then there is no solution for the scheduling problem considered.

2.3 State of the art of intra-piconet schedulings

2.3.1 Knowledge levels

For any global scheduling problem, we can classify existing solutions according to the *knowledge level* they require. The knowledge levels can range from a minimum knowledge to the knowledge of an omniscient observer. These levels define the pertinence and the accuracy of information used by the system. With regard to Bluetooth scheduling, we can define five knowledge levels as described in Table 1. Any knowledge level $0 \leq i < 4$ does not include the message arrival laws.

Table 1. Knowledge levels in Bluetooth.

Knowledge level	**Scale**	**Known features**
Level 0	Per piconet	No information is mentioned System parameters are set to default value for any device within a piconet.
Level 1	Per device	Information given is related to the whole set of messages sent by this device. It is set per device and defined at the link establishment. It does not include the arrival laws.
Level 2	Per class on a device	Information is given per class of messages sent by a device. It is set per class per device and defined at the link establishment. It does not include the arrival laws.
Level 3	Per message flow	The features of each message flow (deadline, size) are known except the arrival law.
Level 4	Omniscient	This level offers the global and complete knowledge of the messages and their arrival times.

For any knowledge level $i, 0 < i < 4$, we can define an associated knowledge level i' including the arrival laws. We notice that level i' provides an estimation of message presence in $S \rightarrow M$ queues more accurate than level i.

2.3.2 State of the art of intra piconet schedulings

Many polling schemes designed or extended for Bluetooth technology have been proposed. They can be classified according to three main criteria.

- **The first criterion allows determining which slaves, among the active ones, are polled**. In *full polling mechanism* such as One-Round Robin or Exhaustive Round Robin, all slaves in the piconet, even if they have nothing to transmit, are polled. In *activity based polling* (e.g. Fair Exhaustive Poller in [5]), the polling cycle can be adjusted according to the activity of each slave. Slaves with empty queues are less frequently polled to prevent the bandwidth wastage.

- **The second criterion corresponds to the minimal knowledge level required to determine the slave polling order**. This order can be determined to achieve fairness access between slaves, to maximize throughput or to improve the adaptability to traffic changes. Static or dynamic priority can be assigned to each slave to determine the polling order. This priority is mapped from information given by the knowledge level. We notice that if a full polling is used, levels 0 or 1 are sufficient as a slave is polled independently of its activity. With regard to intra-piconet scheduling, both criteria lead to the following classification:

- In case of full polling mechanism:
  - If *level 0* is available, the priority is mapped from an arbitrary decision (e.g. arrival order of link requests with 1-RR and ERR in [3]).
  - If *level 1* is available, the slave priority can be mapped from slave parameters:
    - the poll interval T_{POLL} negotiated during the ACL link establishment,
    - the $M \rightarrow S$ queue length (e.g. with EPM in [3]).

- In case of activity based polling:
  - If *level 0* is available, the priority is determined arbitrarily and requires no information (e.g. FEP in [5]).
  - If *level 1'* is available, the priority can be mapped from traffic parameters defined for each slave (e.g. bandwidth and deadline requirements in FPQ in [6]).
  - If *level 2* is available, the slave priority can be mapped from its class parameters:
    - delivery deadline per class,
    - $M \rightarrow S$ or $S \rightarrow M$ queue length (e.g. AFP and SAFP in [7]),
  - If *level 3* is available, the slave priority can be mapped from its message parameters:
    - only the Head-Of-Line message is considered (e.g. PP, KFP in [8]),
    - all the waiting messages are considered.
  - If *level 3'* is available, the message arrival laws can be used to estimate message presence in $S \rightarrow M$ queue (e.g. PFP in [9]) and hence to obtain the slave priority.

- **The third criterion is related to the use of information about the $S \rightarrow M$ queues**. Such a criterion depends on the knowledge level used by the system (as defined in the second criterion). If used, such information is provided by a *signalling scheme* or a *prediction algorithm*. With a signalling scheme for $S \rightarrow M$ queue, slaves can communicate their queue status while transmitting packets. Such information can be included into a specific packet field in $S \rightarrow M$ slot (e.g. [7] uses the flow bit in the header). With a prediction algorithm, the master estimates the message presence in $S \rightarrow M$ queues. This prediction is based on the arrival laws of messages on a slave (e.g. FPQ in [6], PFP in [9]); such prediction mechanism is used with i' knowledge levels.

According to this classification, our CB-EDF solution refers to an activity based polling. The priority of a slave is deduced from its message parameters (class and delivery deadline). CB-EDF uses a combination of signalling and prediction schemes. Unlike other scheduling solutions, CB-EDF accounts for several knowledge levels (precisely 2', 3' and 4). The modularity of CB-EDF allows the selection of the module instantiation adapted to each knowledge level within a unique global solution.

3 Necessity to introduce QoS management

The main problematic of this paper is to study how Bluetooth can support QoS sensitive applications, such as multimedia applications. We first show that 1-RR does not support such applications. We then introduce local QoS management and evaluate the benefit resulting from this enhancement. We then design CB-EDF that supports such applications by including both local and global QoS management. A performance comparison is made by simulations with (i) 1-RR, (ii) 1-RR enhanced with local class management and (iii) CB-EDF with an omniscient knowledge level. We later denote *level 4 CB-EDF* this scheme.

3.1 Support of QoS sensitive applications

We focus on the support of QoS sensitive applications in a Bluetooth network. In particular, we are interested in the QoS requirements expressed by such applications. Two QoS parameters can be defined:

- **The importance degree of a message**. For a given knowledge level, we denote N the number of distinct classes of message provided to the application (with $N \geq 1$) to indicate the importance degree of a message. Messages from class $i+1$ have an importance degree higher than messages from class i.
- **The message delivery deadline**. The application associates with each pending message a relative delivery deadline.

These QoS parameters are independents: a message with a short deadline can have a low importance for the application. Both parameters must be accounted for by scheduling schemes.

3.2 Performance evaluation: Simulation parameters and scenarios

To compare performances of different scheduling schemes, with regard to their QoS support, we run simulations. We have first validated the simulation tool we have developed, by confrontation with already published results [3].

In all the following scenarios, we consider a piconet made up of seven slaves. Four classes are managed (class 0 to 3, class 3 having the highest priority). Each slave generates 2 flows in each of the four classes. In the same way, the master has, for each slave, 4 classes with 2 flows per class. Hence, the ACL uplink and the ACL downlink multiplex eight flows each with different characteristics. Each flow follows a Poisson arrival law. The size of flow messages follows an exponential law of parameter 8, representing the number of slots required for the message transmission. The relative deadlines associated with class 0 and 1 range from 600 to 1000 ms. In class 2, they range from 400 to 800 ms. In class 3, they range from 200 to 400 ms.

Each simulation provides the following results:

- For any class, the *average response time of any flow*. In this paper, we only give the results obtained for the $S \rightarrow M$ flows because they are judged more representative. Indeed, results for these flows are worse than those obtained by $S \rightarrow M$ flows.
- For any class, the *percentage of messages missing their deadline*. For the same reason, we focus on the $S \rightarrow M$ flows.
- The *efficiency of the polling scheme*. It is computed as the percentage of useful slots over the total slot number.

To perform a comparative performance evaluation, we have considered three scenarios representative of various configurations (e.g. process control, centralized application, intra/inter-piconet communications). Traffics are uniformly distributed over the four classes:

- *uniform and symmetric load*: all slaves have the same load and the slave load is equal to the master load for this slave. The submitted load is equal to 96.6%. We denote this scenario by A.
- *uniform and asymmetric load*: the $S \rightarrow M$ load is 3 times higher than the $M \rightarrow S$ load. The submitted load is equal to 87.0%. We denote this scenario by B.
- *non uniform load*: some slaves are heavily loaded. The load of slaves 1 to 3 is three times higher than the load of other slaves. In any case, the $S \rightarrow M$ load is equal to the $M \rightarrow S$ load. The submitted load is equal to 87.6%. We denote this scenario by C.

3.3 Performances of native 1-RR

For 1-RR, the master polls successively each slave for one message according to a fixed cyclic order including all slaves. The average response times for the $S \rightarrow M$ flows are illustrated on Figure 1.

As expected, native 1-RR does not provide service differentiation. Each class has the same average response time. For class 3, the percentage of messages missing their deadline is greater than for other classes in all scenarios (about 11% for scenario A, up to 26% for B and C). It can be explained because the highest priority class has shorter deadlines. The efficiency is about 96.5% for scenario A, 84.9% for B and 84.8% for C.

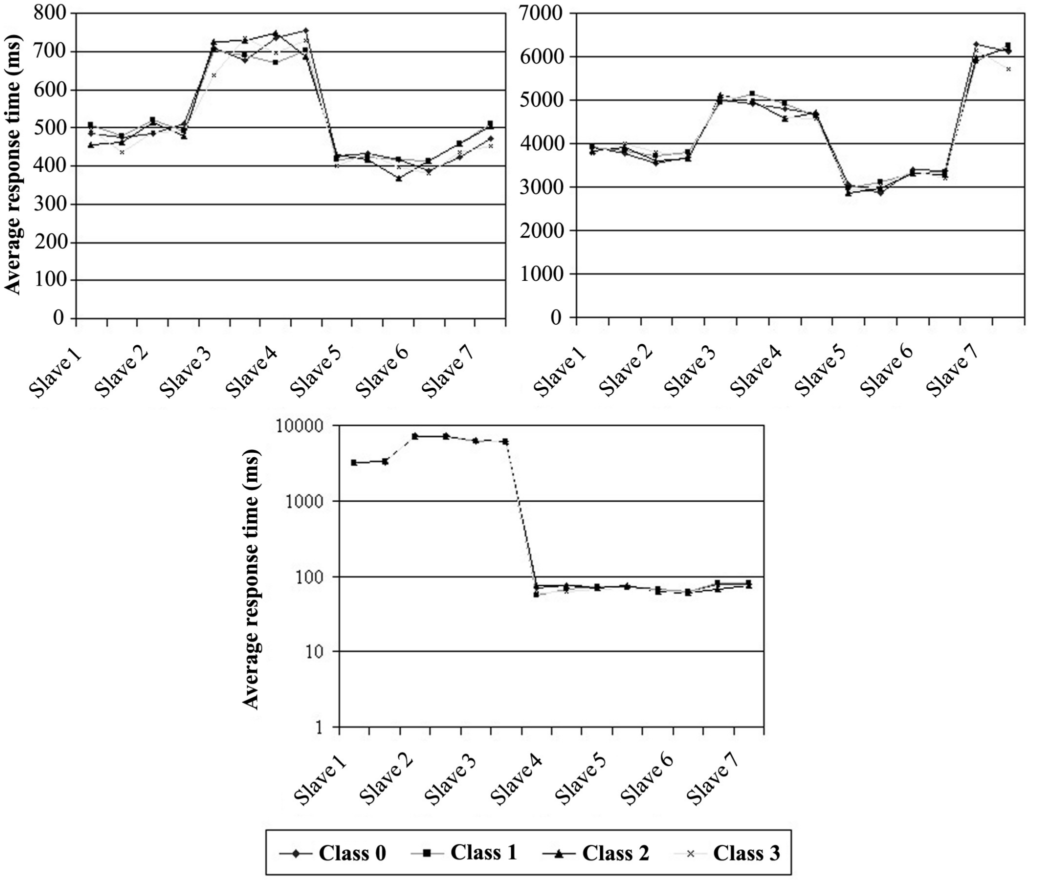

Figure 1: Simulations for 1-RR in scenarios A, B and C.

3.4 Introduction of a local QoS management

3.4.1 Principles

We propose to introduce a local QoS management accounting for the two QoS parameters:

- the importance degree of a message that is mapped in a class,
- the message delivery deadline.

For 1-RR scheme, we enhance its native version with a local QoS management combining class based *Priority Queuing* (PQ) - a class contains all the messages of the same importance degree - and *Earliest Deadline First* (EDF). PQ is used between classes and EDF is used within a class. The intra-piconet scheduling is left unchanged. This local scheduling is used by both master and slaves to manage $M \rightarrow S$ or $S \rightarrow M$ queues. Each $S \rightarrow M$ or $M \rightarrow S$ queue consists of N distinct sub-queues; each class having its own sub-queue. According to PQ principle, each sub-queue has a local priority of transmission. Messages from class $i+1$ are always transmitted before those from class i. In each sub-queue, the transmission order is determined by increasing order of the absolute deadline of each message, according to EDF scheduling. The absolute deadline is equal to the relative deadline plus the arrival time.

This local scheduling will be used with the enhanced version of 1-RR and with our solution CB-EDF.

3.4.2 Performances of enhanced 1-RR

The average response times for the $S \rightarrow M$ flows with enhanced 1-RR are illustrated on Figure 2.

With enhanced 1-RR, we observe that local QoS management introduces an differentiation in case of traffic uniformly distributed over the slaves (scenarios A and B). We notice that, in scenario C, enhanced 1-RR provides to class 3 messages of heavily loaded slaves the same response time as class 0 messages of lightly loaded slaves. The efficiency is the same for 1-RR and enhanced 1-RR.

However, enhanced 1-RR does not guarantee that the messages belonging to the highest priority class are transmitted first.

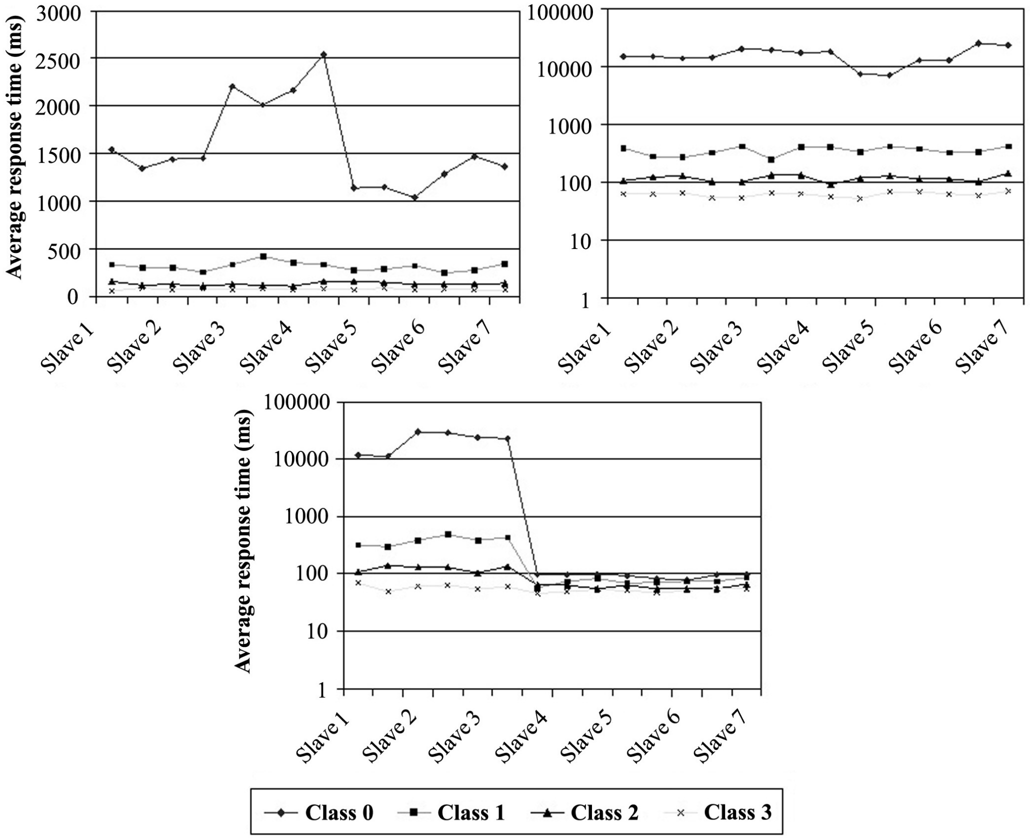

Figure 2: Simulations for enhanced 1-RR in scenarios A, B and C.

3.5 Global QoS management

3.5.1 Principles

Our purpose is to define a global scheduling mechanism that supports service differentiation and accounts for the two QoS parameters. These QoS parameters are accounted for and integrated within Bluetooth local and intra-piconet schedulings by a combination of Priority Queuing between classes and Earliest Deadline First within a class.

That is why we propose CB-EDF. CB-EDF is a modular solution that can be adapted to different knowledge levels ranging from *level 2'* (i.e. information given per class and per device with arrival laws) to *level 4* (i.e. with a complete knowledge).

3.5.2 Functional description

CB-EDF accounts for the class and the absolute deadline of pending messages in the choice of the slave polling order.

First, the master has to consider all $S \to M$ and $M \to S$ pending messages. In *Module 1*, the master determines the message presence in $M \to S$ and $S \to M$ queues of all active slave on the piconet. The module instantiation depends on the knowledge level provided. In considering all $M \to S$ and $S \to M$ pending messages, the master always chooses the slave with the *highest priority class*. In case of equality, the master arbitrates in favor of the slave with the message having the *earliest absolute deadline* in this class.

To apply the PQ principles, the master selects the slave with messages in the highest priority class (*Module 2*). This class is said *eligible*. In case of equality, the master must arbitrate between slaves having messages in the eligible class (*Module 3*). The functional description of CB-EDF run by the master is illustrated by Figure 3. CB-EDF is invoked to select the next slave to poll. It is made up of three functional modules described below.

- **Evaluation of message presence**,
With regard to the knowledge level, the master determines the message presence in $M \to S$ and $M \to S$ queues of each active slave on the piconet.
- **Search of eligible class**,
Searches the eligible class (the highest priority all the active slaves. During the selection process, the master accounts for information from both $M \to S$ and $S \to M$ queues. It selects the highest priority class with pending messages. This class is said *eligible*.
- **Arbitration between slaves having messages in the eligible class**,
Selects the slave having the message with the earliest absolute deadline in the eligible class. This slave is then polled by the master.

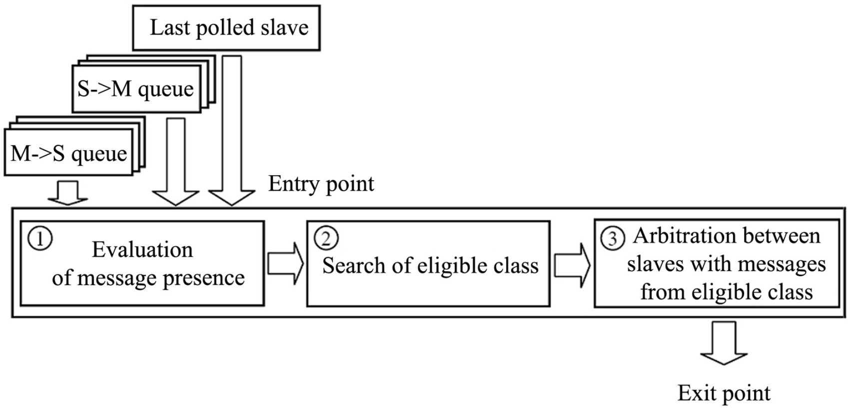

Figure 3: Functional description of CB-EDF algorithm.

3.5.3 Performances of level 4 CB-EDF

In this section, we validate our QoS management principles by evaluating the performances of CB-EDF with an omniscient knowledge level (level 4). In such a case, the master has a global and complete knowledge of the messages and their arrival times in each $M \to S$ and $S \to M$ queues. Unlike other scheduling solutions, this solution allows the master to schedule slaves pending messages in the same way as its own messages. Notice however that level 4 CB-EDF has no practical interest, except for comparative evaluation.

Simulation results, illustrated on Figure 4, show that level 4 CB-EDF provides a very good service differentiation in all the simulated scenarios, better than enhanced 1-RR. Indeed, average response times for the two highest priority classes are shorter than those obtained with enhanced 1-RR. With regard to the percentage of messages missing their deadline, level 4 CB-EDF outperforms 1-RR enhanced solution. In a class, this percentage decreases if the class priority increases (e.g. near to 0% for class 1-2-3, up to 10% for class 0 in scenario A). The efficiency is about 96.4% for scenario A, 85.6% for B and 87.6% for C.

In conclusion, enhanced 1-RR does not succeed in providing a good service differentiation comparatively to level 4 CB-EDF. We notice that enhanced 1-RR performs better when the load is uniformly distributed over the slaves. Hence, these simulation results justify our local and intra-piconet QoS management principles.

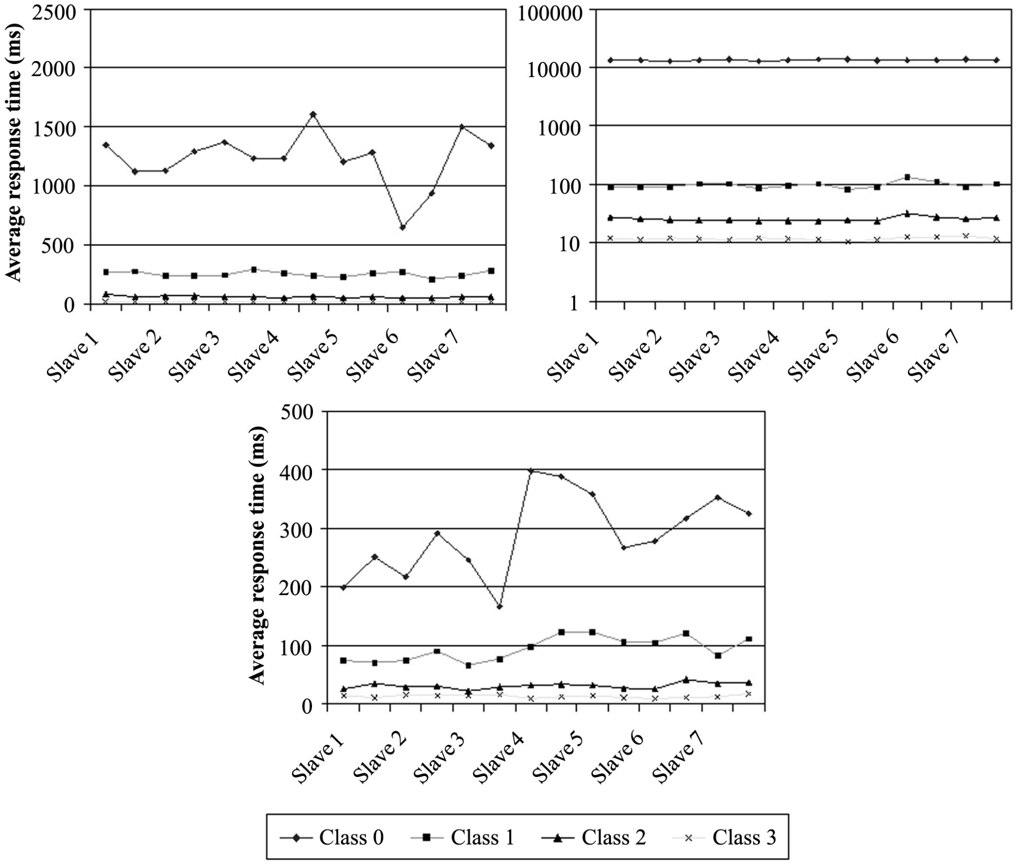

Figure 4: Simulations for level 4 CB-EDF in scenarios A, B and C.

4 Performance evaluation of CB-EDF

As explained in the previous section, level 4 CB-EDF is an unrealistic solution that cannot be implemented. In this section, we show how to implement CB-EDF versions requiring a realistic knowledge level (precisely 2' and 3'). We describe how Module 1 can cope with a knowledge level less than 4. Then, we give performance evaluation of CB-EDF with knowledge level 2'.

4.1 Presentation of CB-EDF

4.1.1 Principles

We defined in section 3 the CB-EDF scheme that globally supports service differentiation and accounts for different knowledge levels. The modularity of CB-EDF allows the selection of the module instantiation adapted to each knowledge level within a unique global solution. The previous description is sufficient for CB-EDF using an omniscient knowledge level (i.e. level 4). For knowledge levels less than 4, module 1, which estimates the message presence, has to account for the following fact: the master cannot have a complete knowledge concerning messages in $S \rightarrow M$ queues unlike messages in $M \rightarrow S$ queues. The knowledge of $S \rightarrow M$ queues is obtained from an original scheme combining signalling and prediction.

Signalling gives information on the instantaneous status of the last polled $S \rightarrow M$ queue. It is used to indicate the presence of pending messages in the same class, or in the strictly lower class, than the last transmitted message. However, this mechanism is unable to signal the presence of messages arrived after the slave poll. That is why the prediction scheme is used to estimate the message arrival in a given class of a slave since its last poll. The combination of signalling and prediction enables to improve the quality of message presence evaluation.

4.1.2 Evaluation of message presence

If knowledge level 3' is provided (resp. level 2'), the master evaluates the message presence in $S \rightarrow M$ queues of each active slave on the piconet, using the message arrival laws of each $S \rightarrow M$ flow (resp. of each $S \rightarrow M$ class) and the signalling scheme.

As we assume that messages in a given class can be generated by different application flows, L2CAP [1] mechanism signalling the presence of another segment in the queue cannot be used. For that purpose, we propose a slight modification in Bluetooth specifications: we use 2 bits (e.g. FLOW and SEQN) in the header of the Baseband packet being transmitted by the slave. If this message, denoted m, is in class c, these two bits are used as follows:

- 00 If $c > 0$, no message of class higher than or equal to $c - 1$ is pending in the $S \rightarrow M$ queue. Otherwise no message is present in any class.

- 01 If $c > 0$, at least one message is present in class $c-1$ with an absolute deadline assumed to be less than or equal to the generation time of m increased by the reference deadline for this class. Otherwise, no message is present in any class.

- 10 At least one message is present in class c with an absolute deadline less than or equal to the absolute deadline of m increased by the reference deadline for this class.

- 11 At least one message is present in class c with an absolute deadline higher than the absolute deadline of m increased by the reference deadline for this class.

For knowledge level 3', the *reference deadline*, defined per class and per slave, is equal to the relative deadline corresponding to the highest arrival rate of message flows. Otherwise, for level 2', the *reference deadline* is equal to the shortest relative deadline of messages in a same class.

As signalling is unable to signal the presence of messages arrived after the slave poll, the master uses a prediction algorithm. This algorithm is based for level 3' on the knowledge of the message arrival law per flow (resp. per class for level 2') to estimate the presence of messages arrived in each class of a slave after its last poll. We assume that message arrival times follow a Poisson process. On a slave, a message is considered present in a class if the probability of having at least one message in this class is higher than or equal to a given threshold. For level 2', the class arrival law is given. For level 3', the arrival law of messages in the considered class follows a Poisson process whose parameter is the sum of the parameters of each flow in the class.

The absolute deadline of this message is assumed to be equal to the first time that gives a probability higher than or equal to the threshold, plus the reference deadline for the message class.

4.2 Performance evaluation of CB-EDF

In this section, we evaluate by simulation the performance of CB-EDF in all scenarios. We consider a 2' knowledge level, i.e. the master only knows arrival laws of messages per class and per slave. Moreover, a threshold of 0.2 is used in the CB-EDF evaluation module. Such a threshold value provides a good estimation of the presence of pending messages in heavily loaded slaves.

Simulation results, illustrated on Figure 5, show that CB-EDF provides a good service differentiation in all the simulated scenarios, better than enhanced 1-RR. Indeed, average response times for the two highest priority classes are shorter than those obtained with enhanced 1-RR. With regard to the percentage of messages missing their deadline, CB-EDF outperforms enhanced 1-RR. In a class, this percentage decreases if the class priority increases (e.g. near to 0% for class 3, up to 10% for class 0 in scenario A).

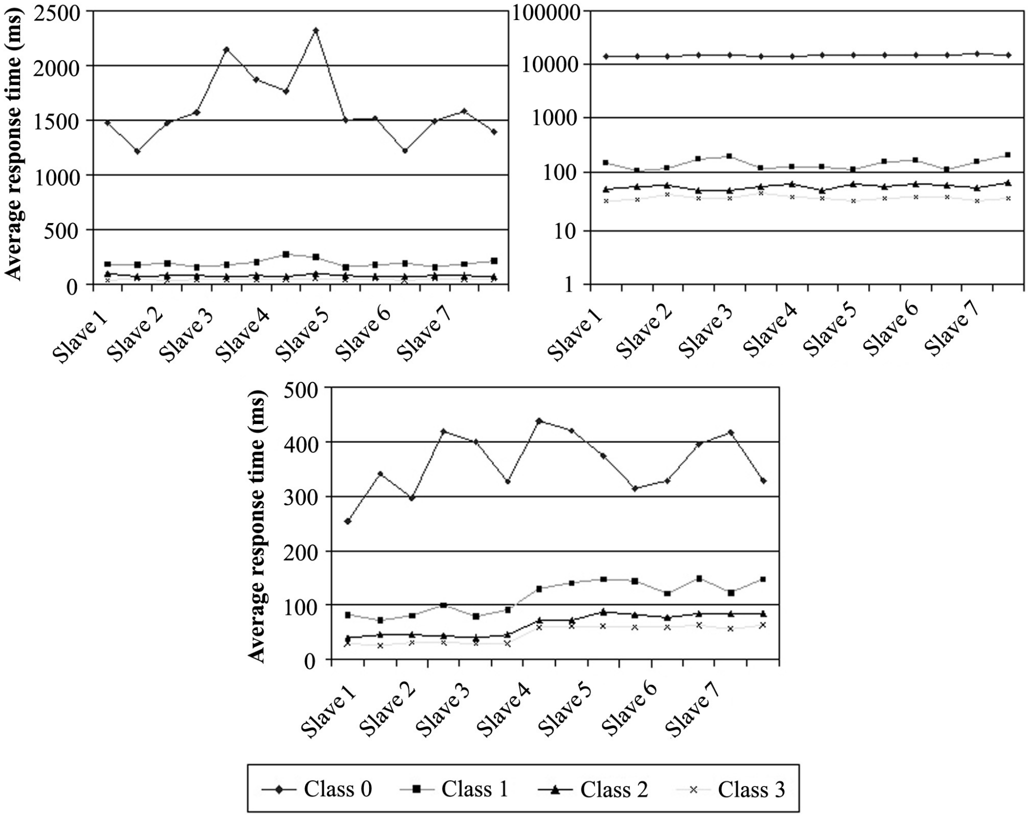

Figure 5: Simulations for CB-EDF in scenarios A, B and C.

In all scenarios, the average response times obtained with CB-EDF are relatively close to those obtained by level 4 CB-EDF. CB-EDF approximates the global exact knowledge of the level 4 solution by combining the signalling scheme with the prediction algorithm. Moreover, we notice in scenario C that heavily loaded slaves have shorter response times than others. Indeed they use the signalling mechanism to notify the master of the message presence. Whereas, the message presence on other slaves is estimated by the prediction algorithm.

The efficiency is the same for CB-EDF and level 4 CB-EDF in all the simulated scenarios (96.4% for scenario A, 85.6% for B and 87.6% for C).

In conclusion, the introduction of classes and deadlines global management in CB-EDF meets the performances expected. Moreover, the average response times for CB-EDF are close to those obtained by the CB-EDF with an omniscient knowledge. This shows the benefit brought by the combination of signalling and prediction algorithms.

5 Impact of the threshold and the knowledge level

In this section, we study more precisely the impact of the prediction scheme, in particular on the average response times of $S \rightarrow M$ messages. We then show the benefit of a higher knowledge level on CB-EDF performances.

5.1 Impact of the prediction scheme on CB-EDF performances

We focus on the impact of the prediction scheme, in particular on the threshold value used to predict the message presence in a given class. On a slave, a message is considered present in a class if the probability of having at least one message is higher than or equal to this threshold. We run scenario B with the threshold value respectively set to the value of 0.2, 0.5 and 0.8. We use CB-EDF with a knowledge level of 2'.

According to the results illustrated by Figure 6, the threshold value has a major impact on average response time (up to +40% higher for class 2 and 3 between simulations with 0.2 and 0.5 values, +45% between simulations with 0.5 and 0.8 values). Moreover, the percentage of messages missing their deadline increases noticeably with the threshold (for class 3, 0% for a threshold of 0.2, 0.25% for 0.5 and up to 2% for 0.8). Indeed, if the probability, given by the evaluation module, of having at least one message in a class has to reach a higher threshold, the interval between two subsequent polls for a same slave increases. In these simulations, the efficiency is left unchanged whatever the case. Simulations show that for high loads, a threshold of 0.2 achieves good performances.

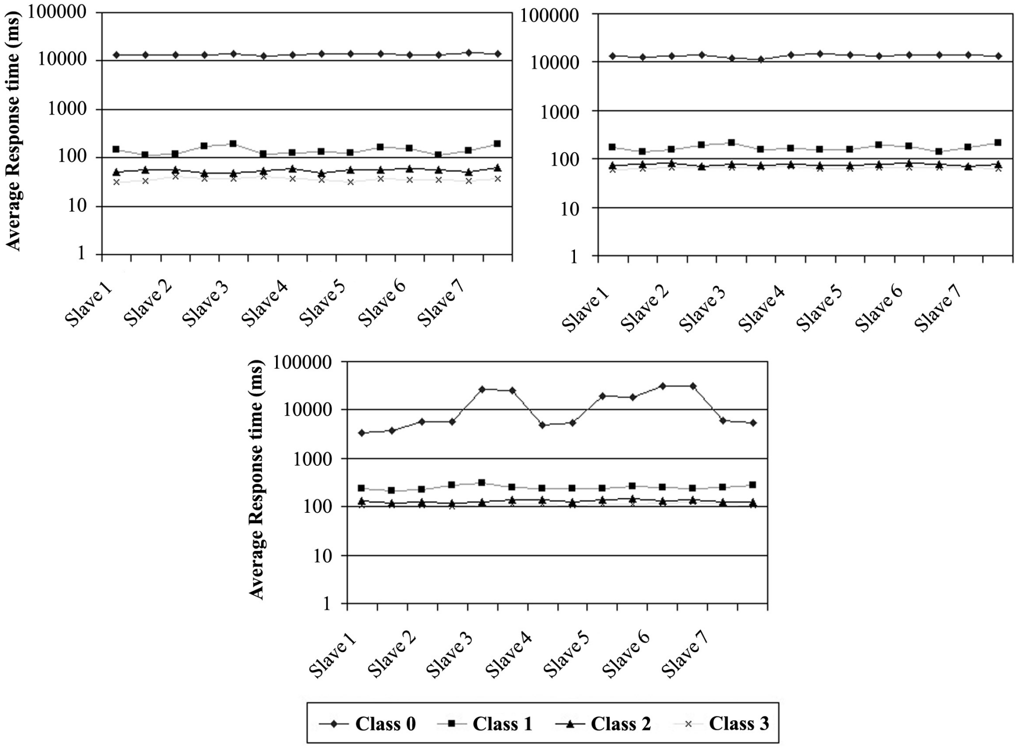

Figure 6: Simulations for CB-EDF in scenarios B with a threshold of 0.2, 0.5 and 0.8.

5.2 Impact of the knowledge level on CB-EDF performances

To show the impact of different knowledge levels on CB-EDF performances, we define a specific scenario where a uniform and asymmetric load is given to all slaves. Unlike scenario B, the two flows in a class have not the same rate for the message arrival law: one rate flow is two times higher than the second one. The submitted load is equal to 93.8%.

Figure 7 shows a slight amelioration if the knowledge level increases from $2'$ to $3'$. In particular, average response times are better for classes 1 to 3 (13% better for class 3, 4% for classes 1 and 2). We notice a decrease of the number of messages missing their deadline whatever the message class. As previously shown, level 4 CB-EDF provides the best performances.

Our simulation results show that CB-EDF exhibits better performances, when the combination of signalling and prediction scheme is more efficient and accurate: if (i) the knowledge level required by the system increases and/or (ii) the threshold value used in the prediction scheme is smaller with high load.

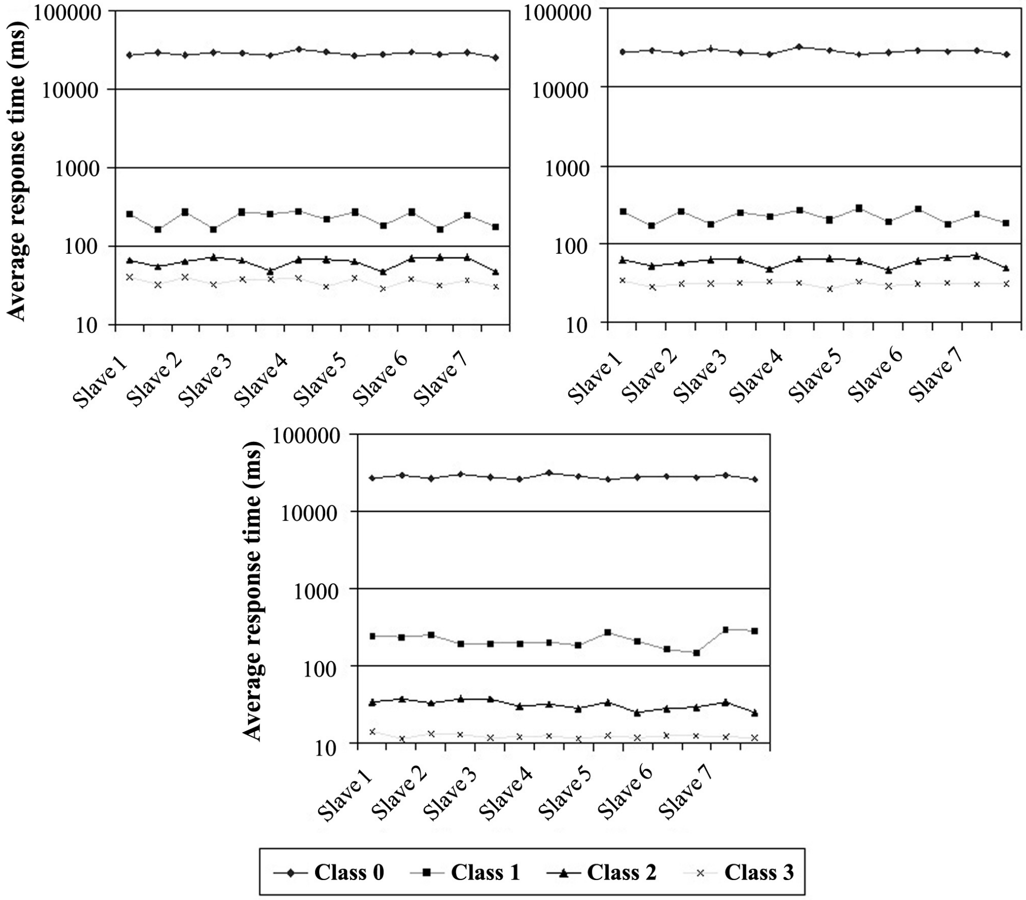

Figure 7: Simulations for CB-EDF in scenarios B with a knowledge level of 2′, 3′ and 4.

6 Conclusion

In this paper, we show how Bluetooth can support QoS sensitive applications. We have considered two QoS parameters defined by the application: the importance of a message and its delivery deadline. The state of the art shows that no Bluetooth scheduling considers simultaneously both parameters. To introduce service differentiation in a piconet, we propose to replace the local scheduling by a combination of class based Priority Queuing and Earliest Deadline First. We show that 1-RR enhanced with this local class management provides insufficient improvements in the support of the two QoS parameters. We then propose to apply these QoS management principles to both local and intra-piconet schedulings: in that purpose we design CB-EDF, a modular solution that can be applied with different knowledge levels. Simulations results show that in all simulated scenarios, CB-EDF achieves a good service differentiation without efficiency loss and outperforms enhanced 1-RR. Moreover, CB-EDF performances increase with the knowledge level. The combined use of signalling and prediction explains why CB-EDF achieves good performances. As a further work, we will extend CB-EDF to inter-piconet scheduling and propose an admission control.

References

[1] The Bluetooth consortium, *Specification of the Bluetooth system, version 1.1*, htttp://www.bluetooth.com, 2001.

[2] A. Mercier, P. Minet, L. George, G. Mercier, *Adequacy between multimedia application requirements and wireless protocols features*, IEEE Wireless Communications, Vol.9, No 6, pp 2-11, 2002.

[3] A. Capone, M. Gerla, R. Kapoor, *Efficient polling schemes for Bluetooth picocells*, Proceedings of IEEE ICC 2001, Helsinki, Finland.

[4] K. Jeffay, D. F. Stanat, C. U. Martel, *On non-preemptive scheduling of periodic and sporadic tasks*, IEEE Real-Time Systems Symposium, San-Antonio, Texas, 1991.

[5] N. J. Johansson, U. Korner, and P. Johansson, *Performance Evaluation of Scheduling Algorithms for Bluetooth*, Proceedings of IFIP TC6 Fifth International Conference on Broadband Communications, Hong-Kong, 1999.

[6] J. B. Lapeyrie, T. Turletti, *Adding QoS support for Bluetooth piconet*, INRIA Research report 4514, 2002.

[7] A. Das, A. Ghose, A. Razdan, H. Saran, R. Shorey, *Enhancing performance of asynchronous data traffic over the Bluetooth wireless ad-hoc network*, Proceedings of IEEE INFOCOM 2001, Alaska, USA.

[8] M. Kalia, D. Bansal, R. Shorey, *MAC scheduling and SAR policies for Bluetooth - a master driven TDD pico-cellular wireless system*, IEEE International Workshop on Mobile Multimedia Communications, p.384-388, 1999.

[9] G. Heijenk, R. A. Yaiz, *Predictive Fair Polling, Provisional Application for United States Letters Patent 60/241*, 314, 2000.

Unité de recherche INRIA Rocquencourt
Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex (France)

Unité de recherche INRIA Futurs : Parc Club Orsay Université - ZAC des Vignes
4, rue Jacques Monod - 91893 ORSAY Cedex (France)
Unité de recherche INRIA Lorraine : LORIA, Technopôle de Nancy-Brabois - Campus scientifique
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex (France)
Unité de recherche INRIA Rennes : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex (France)
Unité de recherche INRIA Rhône-Alpes : 655, avenue de l'Europe - 38334 Montbonnot Saint-Ismier (France)
Unité de recherche INRIA Sophia Antipolis : 2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex (France)

Éditeur
INRIA - Domaine de Voluceau - Rocquencourt, BP 105 - 78153 Le Chesnay Cedex (France)
http://www.inria.fr
ISSN 0249-6399